EDGAR: Energy-efficient Data and Graph Algorithms Research

Aydın Buluç (abuluc@lbl.gov), Ariful Azad
Lawrence Berkeley National Laboratory
https://tinyurl.com/ycvgwogs

Abstract

This project explores methods to increase the energy efficiency of parallel graph and data mining algorithms. We are developing a new family of algorithms that reduce the energy footprint, communication costs, and running time of the graph and sparse matrix computations that form the basis of various data mining techniques. We exploit the well-known duality between graph and sparse matrices to develop communication-avoiding graph algorithms that consume significantly less power. The resulting parallel graph algorithms, including fundamental routines such as graph ordering, matching, and contraction, are scalable beyond thousands of processors.

Motivation

Data are fundamental sources of insight for experimental and computational sciences. The graph abstraction provides a natural way to represent relationships among complex fast-growing scientific data sets. Power consumption is of primary concern on future systems, yet existing graph algorithms consume too much energy per useful operation due to their high communication costs, lack of locality, and inability to exploit hierarchy.

Approach

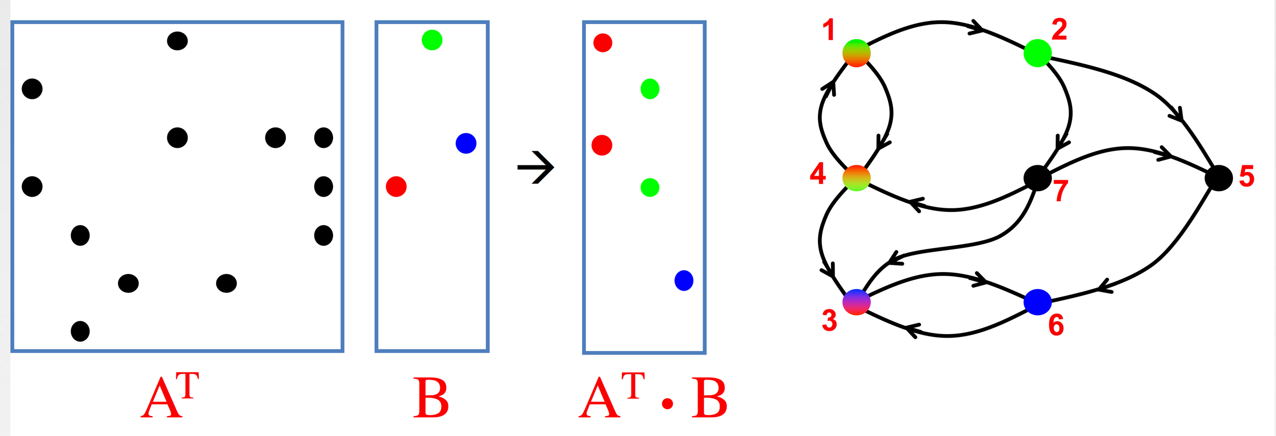
Many graph algorithms have been defined in the language of matrices. We and others have built high-performance graph libraries based on sparse linear algebra. The PI and collaborators formed the GraphBLAS Forum (http://graphblas.org) to standardize the low-level building blocks used in graph algorithms.

- Completed the mathematical formalizations of GraphBLAS.
- Defined the binding of the C programming language onto the mathematical definition, creating the GraphBLAS C API.
- Our Berkeley team performed fundamental research in defining the functions and their mathematical semantics.
- Worked on efficiently mapping problems that are of interest to DOE onto GraphBLAS (see Ariful Azad's poster for a comprehensive list)
- Developed novel communication-avoiding parallel algorithms for the GraphBLAS functions.

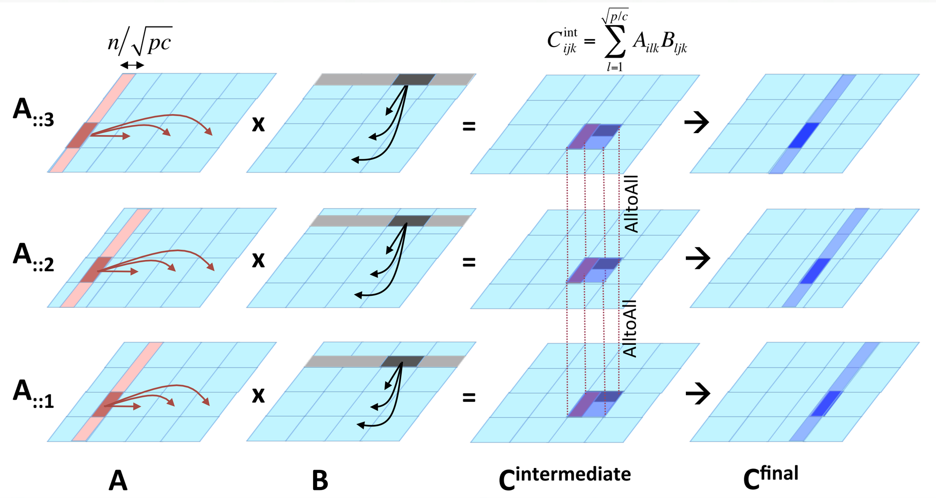
Additional research is needed to develop communication-avoiding and work-efficient algorithms for the complete GraphBLAS specification in its full generality.

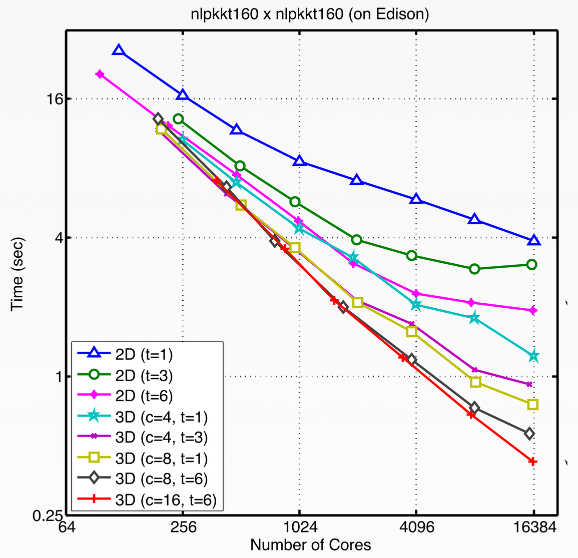
Major results

- GraphBLAS mathematical description [1] and the C API specification [2]

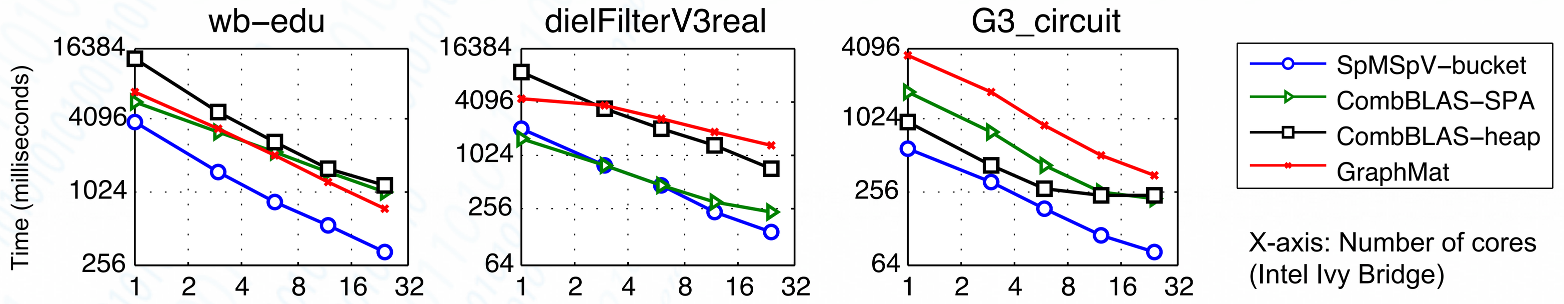


- Parallel algorithms for sparse-matrix- sparse matrix multiplication (SpGEMM) [3]
  1. Split-3D-SpGEMM: an efficient implementation of communication-avoiding SpGEMM [4]
  2. Novel shared-memory kernel for in-node parallelism [4]

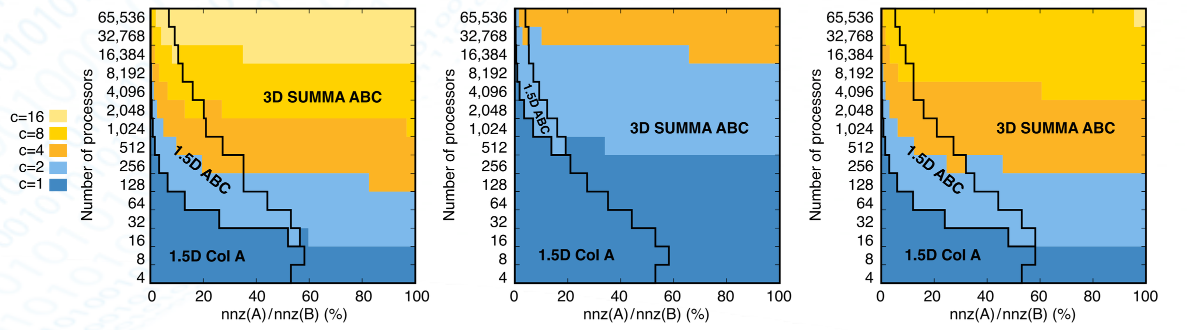




- A work-efficient parallel algorithm for sparse matrix-sparse vector multiplication (SpMSpV) [5]
  1. First ever work-efficient algorithm for SpMSpV: 15x speedup on 24 cores & up to 49x speedup on 64 cores.
  2. Up to an order of magnitude faster than its competitors, especially for sparser vector



- Communication-avoiding algorithm for sparse-dense matrix multiplication [6]. Explored all possible variants of algorithms, proved tight lower bounds, identified the best algorithm for each sparsity and concurrency regime.



Used in partial correlation estimation and dimensionality reduction

Conclusions and Future Work

Graph abstraction is a powerful way of organizing and representing data; hence making graph algorithms ubiquitous in data analytics.

Big idea: sparse matrix-graph duality makes graph algorithms scalable and allows us to avoid communication at the sparse matrix algebra level. Algorithms that avoid communication are also energy efficient.

The graph algorithms enabled by EDGAR are often used as building blocks of algebraic solvers, such as direct and iterative sparse linear solvers and algebraic multigrid. These solvers are heavily used by DOE Office of Science researchers, where we anticipate immediate impact.

On longer time frame, the primitives developed here will have impact on bioinformatics, data mining and analysis, machine learning, and electronic structure calculations.

Areas in which we can help

- Scalable Machine Learning
- Extreme-Scale Data Analysis
- Biological Systems Science
- Compiling and Scheduling for Quantum Computing
- Sparse Numerical Solvers

Areas in which we need help

- Domain science expertise for interpreting results of methods and algorithms we develop.
- Lightweight parallel programming abstractions for productive and high-performance programming.

References

1. J. Kepner, P. Aaltonen, D. Bader, **A. Buluç**, F. Franchetti, J. Gilbert, D. Hutchison, M. Kumar, A. Lumsdaine, H. Meyerhenke, S. McMillan, J. Moreira, J. Owens, **C. Yang**, M. Zalewski, T. Mattson. *Mathematical foundations of the GraphBLAS*. In IEEE High Performance Extreme Computing (HPEC), 2016
2. **Aydin Buluç**, Timothy Mattson, Scott McMillan, Jose Moreira, and **Carl Yang**. Design of the GraphBLAS API for C. In *IEEE Workshop on Graph Algorithm Building Blocks, IPDPSW*, 2017.
3. Grey Ballard, **Aydin Buluç**, James Demmel, Laura Grigori, Benjamin Lipshitz, Oded Schwartz, and Sivan Toledo. Communication optimal parallel multiplication of sparse random matrices. In *SPAA 2013: The 25th ACM Symposium on Parallelism in Algorithms and Architectures*, Montreal, Canada, 2013.
4. **Ariful Azad**, Grey Ballard, **Aydin Buluç**, James Demmel, Laura Grigori, Oded Schwartz, Sivan Toledo, and Samuel Williams. Exploiting multiple levels of parallelism in sparse matrix-matrix multiplication. *SIAM Journal on Scientific Computing (SISC)*, 38(6):C624-C651, 2016.
5. **Ariful Azad** and **Aydin Buluç**. A work-efficient parallel sparse matrix-sparse vector multiplication algorithm. In *Proceedings of the IPDPS*, 2017.
6. Penporn Koanantakool, **Ariful Azad, Aydin Buluç**, Dmitriy Morozov, Sang-Yun Oh, Leonid Oliker, and Katherine Yelick. Communication-avoiding parallel sparse-dense matrix-matrix multiplication. In *Proceedings of the IPDPS*, 2016.